Data Analysis and Visualization in R (IN2339)

Exercise Session 9 - Statistical Assessments for Big Data

Jun Cheng, Christian Mertes, Vicente Yépez, Julien Gagneur

Section 00 - Getting ready

```r
library(ggplot2)
library(data.table)
library(magrittr)
library(tidyr)
library(dplyr)
library(patchwork)
```

Section 01 - Quantile-Quantile plots

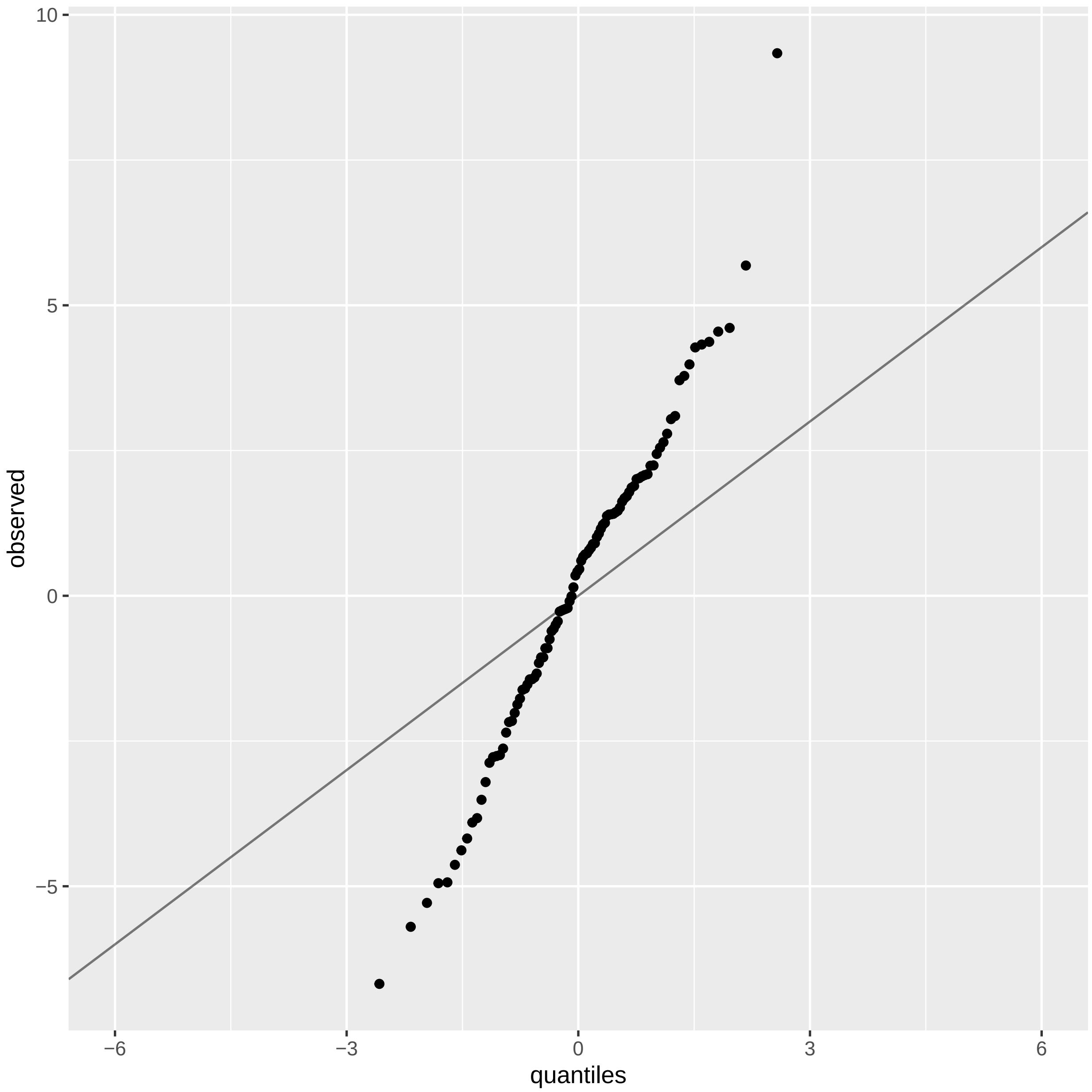
We will simulate some data from different distributions and will compare their quantile-quantile plots.

1. We will use a standard normal ($\mu = 0$ and $\sigma^2 = 1$) distribution as a reference set. Please create a data table with 100 observations with a column containing the simulated observed values. Plot first a histogram of the observed values. Then create a column containing the expected quantiles and plot the expected against the observed quantiles.

Hint: Set the x and y limits to `[-6,6]` where appropriate.

2. Now add a normal distribution with $\mu = 4$ to your `data.table` and plot the Q-Q plot. How did it change?

3. How would you tweak the distribution so that you get the following Q-Q plot?



Section 02 - QTL mapping of growth

For the next questions, we will use the yeast dataset.

```
library(tidyr)
library(data.table)
library(ggplot2)
library(ggthemes)

## load the data
genotype <- fread("extdata/eqtl/genotype.txt")
growth_rate <- fread("extdata/eqtl/growth.txt")
marker <- fread("extdata/eqtl/marker.txt")
```

```
setnames(marker, 'id', 'marker')
genotype <- genotype %>%
  melt(id.vars = 'strain', variable.name = 'marker', value.name = 'genotype')
```

1. Test for markers associated with growth.

Look for a genetic marker associating with growth rate in maltose. To do so, run a wilcoxon test for growth rate versus the genotype at each of the 1,000 markers. Remember that we tested this relationship for one specific marker last time. Plot a histogram and a Q-Q plot of the obtained p-values. Which ones would you consider significant and why? Do we need to correct for multiple testing?

Hint: plot p-values in `-log10` scale.

2. Plot the p-values against genomic position. Do you see positions that associated with growth? The genomic position is defined by the chromosome the marker is on and the marker's position within that chromosome.

Hint: plot p-values in `-log10` scale. Use `start` column for position from the marker table. In ggplot, use facet on chromosome.

3. How many marker significantly associate with growth after correcting for multiple testing? Find the two markers with the lowest p-values.

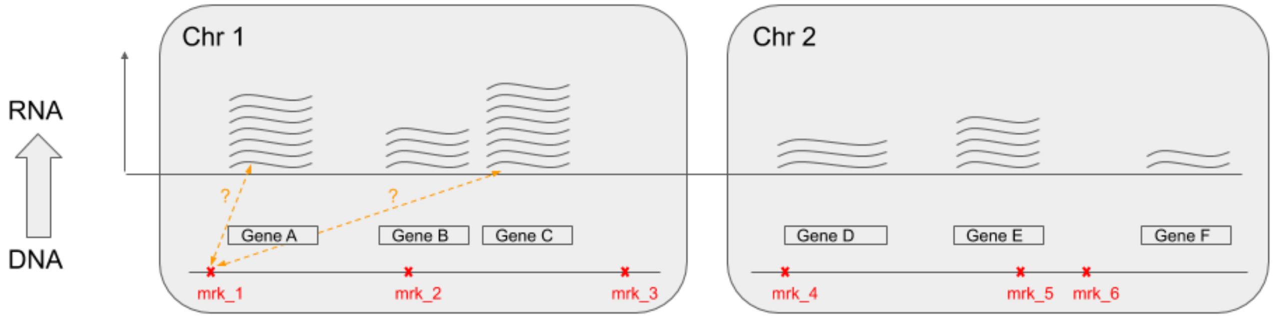
Section 03 - QTL mapping of gene expression (eQTL)

In this exercise, we move to a much larger dataset. RNA abundances have been measured for 8,382 genes in different strains of yeast grown in different environments.

Gene expression

The overall fitness, for example the growth rate of a micro-organism, is the outcome of complicated molecular processes occurring in cells. One of the most fundamental molecular process is called transcription: it is the production of RNAs for a given gene.

Transcription is a quantitative process. Cells can produce varying amounts of RNAs for a given gene depending on the environment or the genotype.



Assessing associations of the strains genotype and their RNA abundance leads to a massive amount of tests.

Install library `genefilter`

We start by installing **`genefilter`**, an R package which among other things allows fast iteration of testing with the T-test over rows of a matrix. This call will prompt the question "Update all/some/none? [a/s/n]:", answer none (press 'n').

```
#install.packages(’BiocManager’)
#BiocManager::install("genefilter")
library(genefilter)
```

Gene expression data

The expression data records the gene expression of 8,382 genes, of different strains in different environments. The units are arbitrary.

```
expression <- fread("extdata/eqtl/expression.txt")
# subset to the strains grown in YPMalt.
expression <- expression[ , grep(’YPMalt’, colnames(expression)), with=FALSE]
# rename columns to only identify the strain.
colnames(expression) <- sapply(strsplit(colnames(expression), ’-’), function(x) x[2])

expression[, c(1:8)]
```

Prepare the genotype matrix

```
# Prepare the genotype matrix for eQTL testing. This time it is better to have it
# in a wide format. So let’s just read it in again.

genotype <- fread("extdata/eqtl/genotype.txt")
# subset the genotype matrix to the samples for which expression data exists.
genotype_expression_profiled <- genotype[strain %in% colnames(expression)]
# drop the strain column.
genotype_expression_profiled <- genotype_expression_profiled[,-1]
genotype_expression_profiled[1:5, 1:5]
```

Gene expression profiling at 8,382 genes (including coding and non-coding RNAs) has been performed for 34 strains grown in the same media. The log-expression levels are stored in the matrix ’expression’, the genotype information for these 34 strains are in the ’genotype_expression_profiled’ data table.

Test and plot the significant associations between genotype and gene expression $FDR < 0.1$.

First, obtain a matrix (# of genes, # of markers) containing all p-values from t-tests between gene expression and genotype. Then, correct it for $FDR < 0.1$ and obtain the significant ones. Scatterplot only the significant associations. Hint: Use the function `rowttests()` from the ’genefilter’ package which fastly applies a t-test to each row of a matrix (`as.matrix()`).

OPTIONAL Section 04 - P-values and FDR

Here we will use simulations to investigate the effect of sample size and of the proportion of true and false null hypotheses when performing multiple testing. We will do it for the problem of two-sample comparison with equal sizes.

We are interested in comparing the observations of two samples: $x_1, ..., x_n$ and $y_1, ..., y_n$. Specifically, we ask whether the expectations differ using a two-sample Student t-test.

1. simulate data under the null hypothesis $H_0 : \mu_x = \mu_y$.

We simulate $N = 10,000$ times two samples $x_1, ..., x_n$ and $y_1, ..., y_n$ where X and Y follow the standard normal distribution. We use sample size $n = 50$ but our function works for any `sample_size`. For each simulated dataset, we compute the two-sided p-value of a t-test. We assume unequal variance as by default in the R function `t.test()`.

You can use the following function to do this:

```r
simulate_norm_null <- function(sample_size=50, N_experiments=10000){
  sapply(seq(N_experiments), function(i){
    x <- rnorm(sample_size)
    y <- rnorm(sample_size)
    t.test(x, y, alternative="two.sided")$p.value
  })
}
```

2. Plot the p-values simulated before.

How are the p-values distributed? Create a function that given a vector of p-values, plots a histogram of them. Plot p-values for $sample\_size = 50$. What could be a better visualization?

3. compute the quantiles and add a Q-Q plot to the histogram.

If all tests are truely under the null hypothesis, the distribution of the p-values should be uniform by definition. A quantile-quantile plot compares the expected p-values with the observed onse. What are the quantiles for p-values and how can we compute them? Add the Q-Q plot to the function from the last question and add a line where you expect the points to be. Please plot again the p-values for $sample\_size = 50$ with the new function.

Hint: add `title` as a parameter for later.

4. Correct for multiple testing

Adjust p-values with the different methods seen in the class. Plot the results using the plot function. Do they behave as expected? Discuss.

5. Simulate data under the alternative hypothesis $H_1 : \mu_x \neq \mu_y$

Simulate $N = 1,000$ times two samples $x_1, ..., x_n$ and $y_1, ..., y_n$ where X and Y follow the normal distribution with $\mu_x = 0$ and $\mu_y = 0.5$ with a $sample\_size = 10$. For each simulated dataset, compute the two-sided p-value of a t-test. You can assume unequal variance as by default in the R function `t.test()`. Create the same p-value plots as done before.

OPTIONAL Section 05 - sample size and power

1. Investigate the effect of different sample sizes n (10, 100, 1000) on the p-value plots of the question above. Discuss.

2. p-values for a mixture of null and alternative.

Provide the same plots as before when considering a dataset of $N_0 = 10000$ data points simulated under H_0 (true null) and $N_1 = 1000$ data points simulated under H_1 (false null). Discuss. For p-values one is mostly interested in the lower range. Think of a transformation on how to visualize the p-value data in a better form and change the plotting function accordingly.

3. Mixture of H_0 and H_1 adjusted for multiple testing

Adjust the p-values with Benjamini-Hochberg (FDR) in the mixture from the previous question. Make a contingency table of true positives, true negatives, false positives and false negatives. Try this with different sample sizes for $\text{FDR} = 0.05$. Discuss.

Do the same thing for the bonferroni correction and compare the results

Hint: Create a function with at least `sample_size` as parameter. Use `names(pvals) <- rep(c("H0", "H1"), c(10000, 1000))` to name your p-values. You can then use `table` to create your contingency matrix.